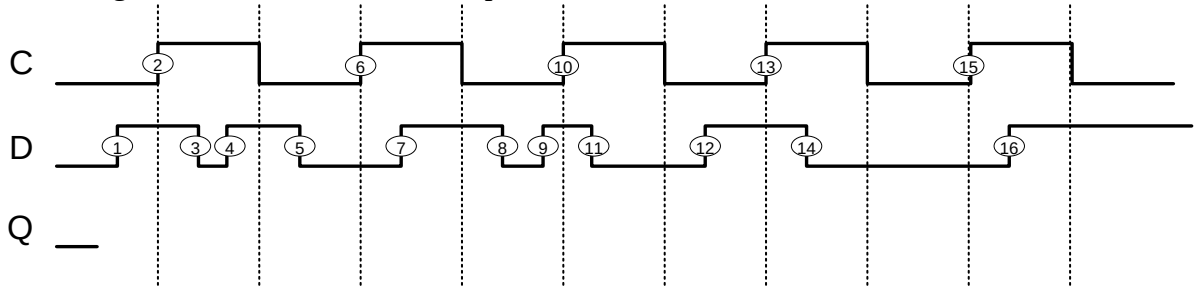
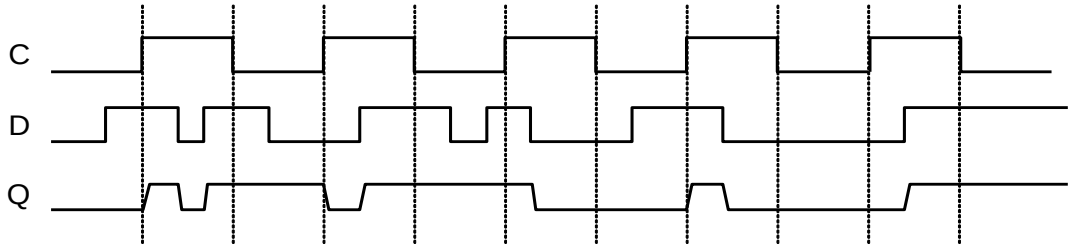
EE 109 Homework 6

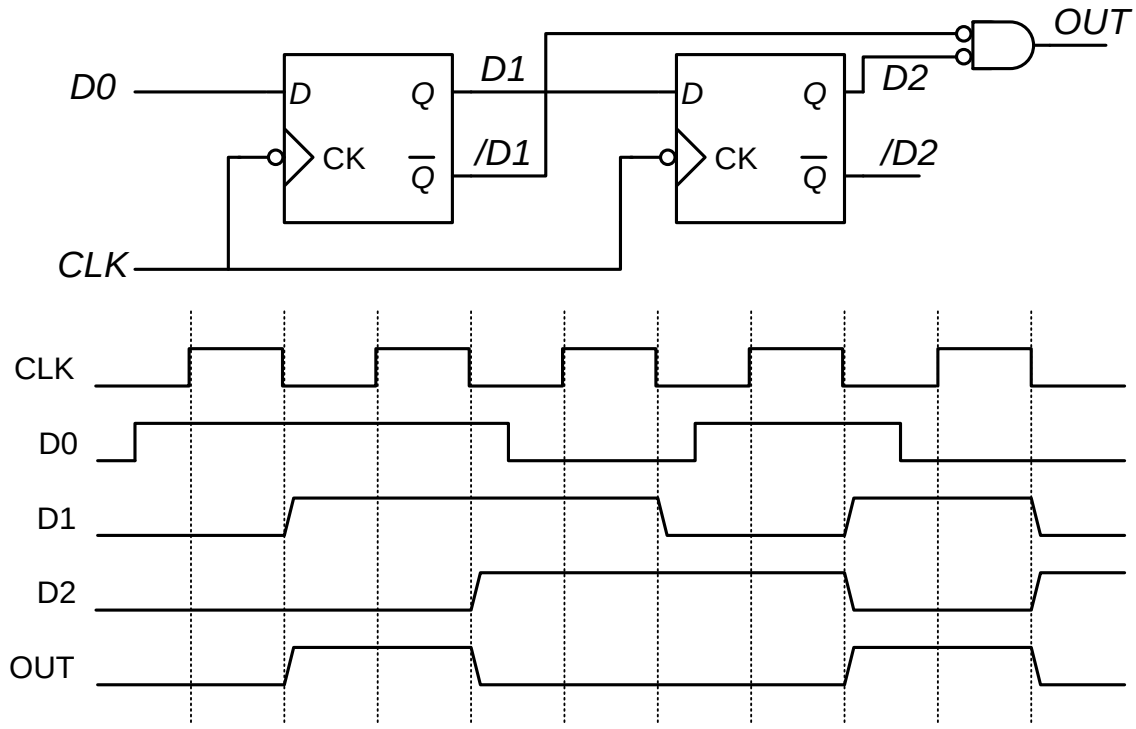
Name: __Solutions_____
Due: **Score: _____**

1. [BB] Complete the following waveform for a D-Latch with active-high clock (Q responds to D when C=1). Remember latches are level sensitive. To enter your answer online, select which transitions of the clock, C, or the D-input (indicated by a timestamp) will cause Q to toggle (i.e. change from 0 to 1 or 1 to 0). If Q doesn't change do not enter that timestamp as an answer.



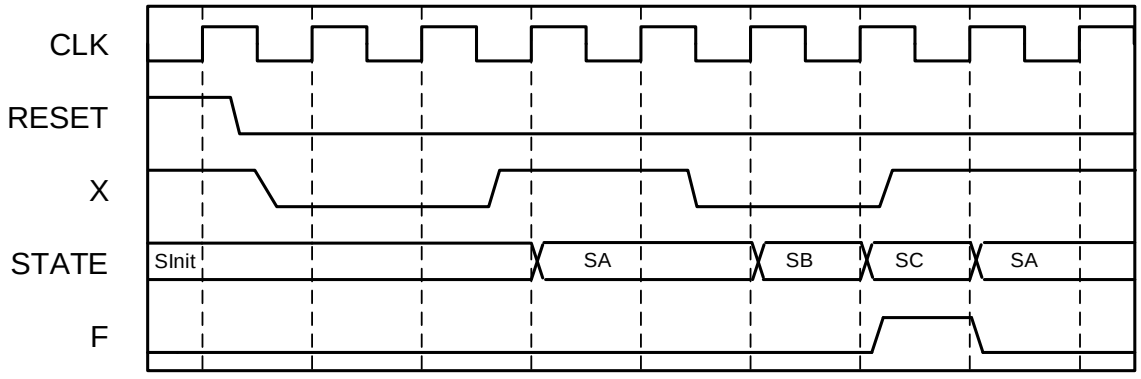


2. Complete the waveform for the following design involving two negative edge-triggered D flip-flops. Note: OUT is a combinational logic function of the FF outputs. Note: Combinational logic gates are UNAFFECTED by the clock (only FF's use the clock signal). Thus, a change in the inputs to logic gates causes an immediate change (after a small propagation delay) in the outputs. For D1 and D2 indicate their value during the middle of clock cycles A through F. For OUT indicate which timestamps of D or CLK will cause (either directly or via D1 and D2) OUT to toggle (from 0 to 1 or 1 to 0)



3.

The waveform is below:



The circuit is checking for 1001 (1 moves to SA, 0 moves to SB, another 0 moves to SC and then X has to be 1 to make F=1)

4.

(a) There is only one D Flip flop, and Q_1 can be 0 or 1. Thus, there are 2 states for this circuit.

(b) The maximum number of transition arrows originating from one state is 8. Because we have A, B and C three external inputs, and there are 8 possible combinations of one 3-bits binary number.

(c) No. Because when A is equal to 1, the input of the D Flip flop is definitely 0. Thus, when the next clock tick comes, Q_1 is equal to 0 and F is 0 too.

5.

Using 1-hot state machine design, we use 1 flip flop per state: QOff, QMon, Q1wr, QAlarm and the corresponding D inputs. The inputs are E=ENTER and C=CORRECT.

Next State Logic

We can arrive at the next state equations by inspection from the state diagram (looking at each transition pointing TO a state.

DOff = Qoff & (~Enter + ~Correct) + (QMon + Q1wr) & Enter & Correct
// can be ~(Enter & Correct)

DMon =Qoff & Enter & Correct + QMon & ~Enter
D1wr = QMon & Enter & ~Correct + Q1wr & ~Enter
DAlarm = QAlarm + Q1wr & Enter & ~Correct

Output Logic

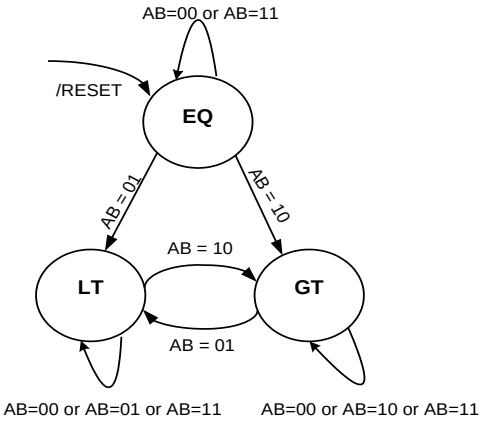
Alarm = QAlarm

Reset Logic

	Initial Value	CLR	SET
QOff	1	GND	RESET
QMon	0	RESET	GND
Q1wr	0	RESET	GND
QAlarm	0	RESET	GND

6. (a.)

(1) First, we build the state diagram.



Next State Logic

We can arrive at the next state equations by inspection from the state diagram (looking at each transition pointing TO a state.

DEq = QEq & (~A&~B + A&B)
// can be QEq & (A xnor B)

DGt = (QEq + QLt)(A & ~B) + QGt & ~(~A&B)
DLt = (QEq + QGt)(~A & B) + QLt & ~(A&~B)

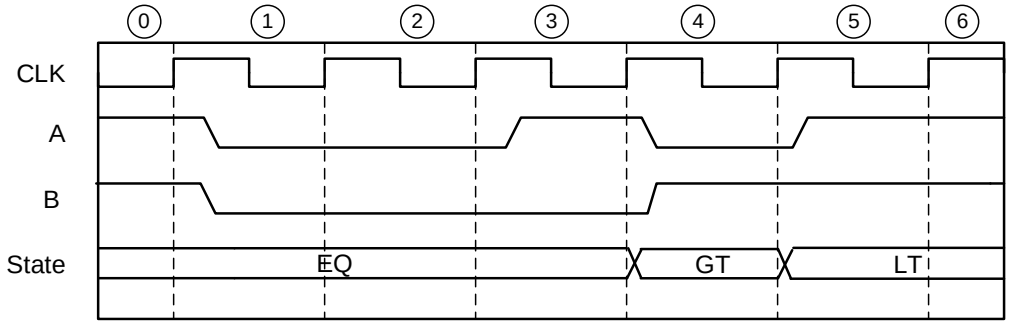
Output Logic

Eq = QEq
Lt = QLt
Gt = QGt

Reset Logic

	Initial Value	CLR	SET
QEq	1	GND	RESET
QLt	0	RESET	GND
QGt	0	RESET	GND

The waveform is below:



7. Complete the waveform for a 4-bit counter with count-enable and data load.